

## Identifying Persistent X States in Serving RAM's Read Path

This analysis focuses on an early-stage waveform captured during simulation of an SRAM module with a Wishbone interface. The waveform highlights a critical issue: persistent X (undefined) values on the `o_rdata` signal for multiple cycles during read operations. This behavior can lead to non-deterministic data reads and violates expected timing behavior for memory access. By examining this waveform, we pinpoint the underlying causes and their ripple effects on the bus signaling and output data.

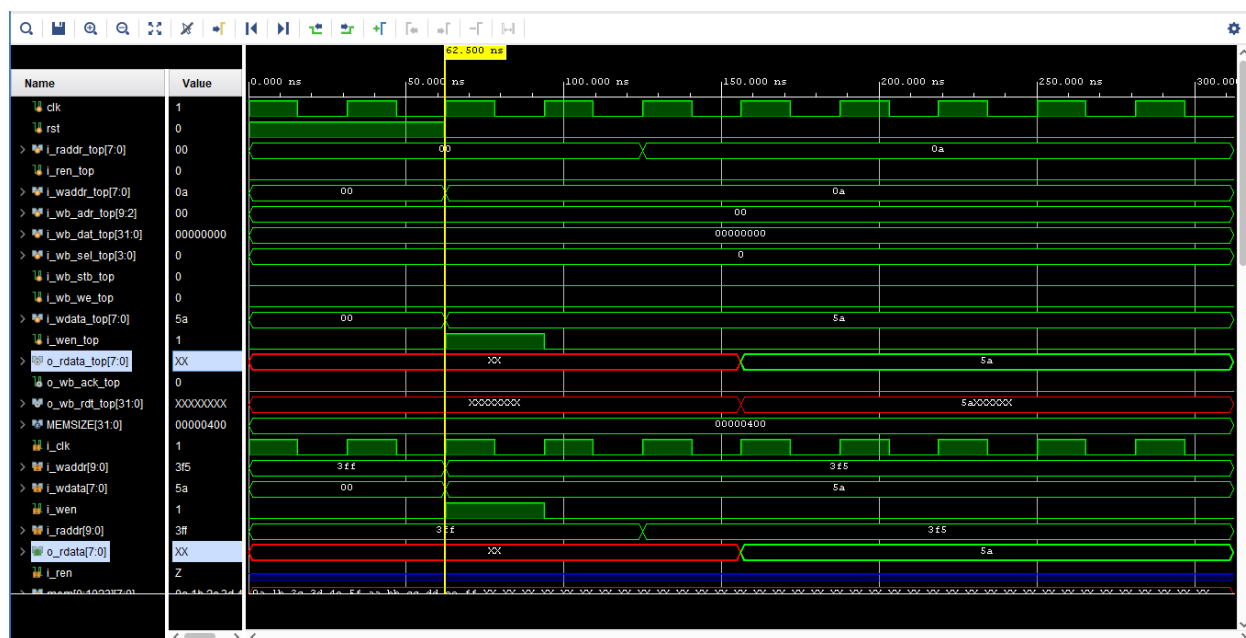


Figure: 01 – Faulty Read Behavior – Baseline Waveform for Debugging

### Analysis of the Waveform (Image Above)

This waveform captures faulty behavior in the SRAM's read path. Specifically:

- `o_rdata_top[7:0]` and `o_rdata[7:0]` stay X for an extended duration, even after valid address and enable signals are applied.
- A read request is made (`i_ren_top = 1`, `i_raddr_top = 0x00`) while the system is out of reset, yet the read data output is not valid for **multiple clock cycles**.
- `o_rdata` is not driven immediately and stays undefined. This suggests either the internal memory was not initialized or the logic to drive `o_rdata` was gated behind handshake signals.
- Meanwhile `i_wdata_top`, `i_waddr_top`, etc. show correct behavior, confirming the write path works fine.

## Core Problem

Figure 01 represents a **functional issue** — not just a transient glitch. The prolonged undefined state of `o_rdata` makes it unreliable during the first few read cycles. The root cause lies in the lack of:

- Proper memory initialization,
- Default assignments to output signals,
- Immediate data forwarding logic upon valid reads.

This behavior must be corrected to ensure timing-accurate and deterministic reads from the memory module.

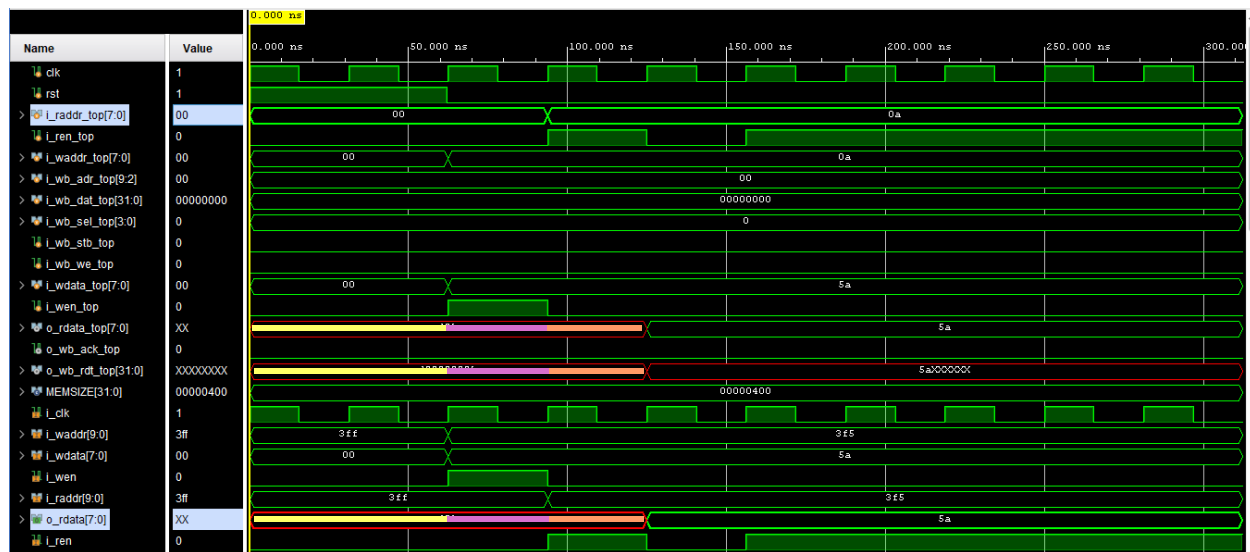


Figure: 02 – Regions of fault identified

### Yellow Region: Reset is High

In the yellow-highlighted segment at the beginning of the simulation, the `i_rst` signal is asserted (1). During this phase:

- Although the clock is toggling, the design is **logically in reset**.
- Even if the `serving_ram` module does **not contain explicit reset logic**, it is being driven by signals from `servile_rf_mem_if` which is sensitive to reset.
- Since the memory isn't being read or written (due to the system still in reset), and because `o_rdata` is not explicitly assigned during reset, it remains **undefined (X)**.

### ■ Pink Region: Write Enable (i\_wen) is High

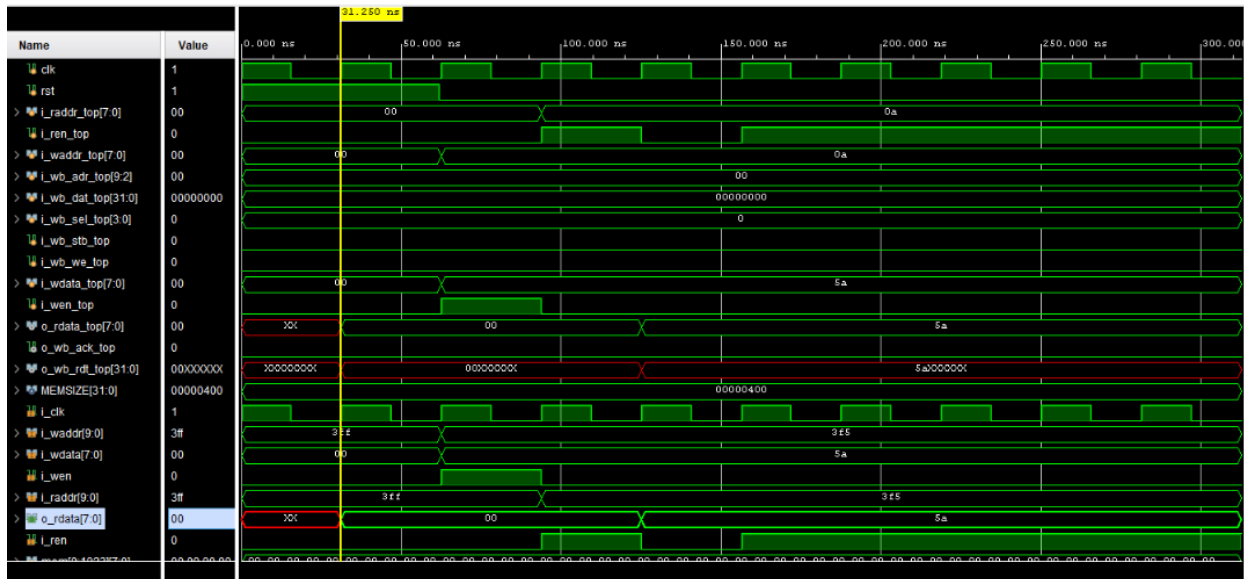
In this highlighted segment, you observe that i\_wen is high, indicating a **write operation** is taking place. During this time:

- The RAM is writing i\_wdata to the location specified by i\_waddr.
- However, in the original implementation of serving\_ram, the o\_rdata register is **only updated when i\_wen is low** (i.e., during a read).
- Since o\_rdata is not assigned during writes, its value **lingers in an undefined state** unless previously defined.

### ■ Orange Region: Address Setup Violation

This region highlights a **timing issue** — the read address i\_raddr is not stable before the rising edge of i\_clk.

- RAM typically requires **address setup time**, meaning the address should be stable for a short duration **before the clock edge**.
- In this case, i\_raddr is changing **exactly at or just before** the rising edge.
- This means the value latched by the RAM could be unpredictable or lead to undefined behavior.
- The result is that the mem[i\_raddr] access in the RAM returns an X, which is then assigned to o\_rdata.



**Figure: 03** – Response after memory initialization

```
INFO: [XSIM 43-4323] No Change in HDL. Linking previously generated obj files to create kernel
INFO: [USF-XSim-69] 'elaborate' step finished in '1' seconds
Time resolution is 1 ps
Running RAM translation/access test...
[READ TEST] Addr: 0x0a, Expected: 0x5A, Got: 0x5a
```

The waveform shows the operation of a simple synchronous RAM module. Initially, o\_rdata is undefined (X) because it hasn't been assigned a value before the first clock edge. After the rising edge of i\_clk and deassertion of i\_rst, a write occurs (i\_wen=1, writing 0x5a to address 0x3f). On the subsequent read (i\_wen=0), o\_rdata correctly outputs 0x5a, confirming successful memory write and readback.